## REMARKS

In view of the foregoing amendments and following remarks responsive to the Final Office Action dated December 12, 2006, Applicant respectfully requests favorable reconsideration of this application.

The Office had previously rejected all claims, claims 1-33 as obvious over Howard in view of Sears. Applicant respectfully thanks the Office for withdrawing these previous rejections.

However, in this latest Office Action, the Office has rejected all of claims 1-33 as obvious over Howard in combination with Sears and the newly cited JSOnline reference.

Applicant respectfully traverses.

### Introductory Comments

The present rejections are very similar to the previous rejections asserted in this case. The Office has added the new JSOnline reference to overcome one aspect of Applicant's previous arguments.

The present situation is not one in which one or a few claim elements are not found in the prior art of record or in which there is a subtle issue as to the propriety of a proposed combination. In the present situation none of the references, including the newly cited JSOnline reference, discloses any of the teachings for which they have been cited. In fact, none of the references even pertains to the subject matter of the

present invention, namely, synchronizing the same cookie across a plurality of client machines used by a single user entity.

The differences between the present invention and the prior art are so significant and numerous that Applicant provides below a step-by-step detailed analysis of each claim element of exemplary claim 1 and the prior art that the Office has asserted as disclosing that element, including a detailed discussion of why each one of them is not found in the prior art of record.

**The Present Invention**

The present invention is a method for synchronizing copies of the same cookie(s) across a plurality of different client computers of a single user. Particularly, it is not uncommon for a single person to have multiple computers that he or she uses on a regular basis for accessing the Web.

In accordance with the present invention, such a user registers with a service that will synchronize the copies of the cookies across all of that user's computers. Particularly, a user opens an account and registers all of his or her computers under that account. A server (the Cookie Synchronization Server) stores the cookies and/or cookie change information of its registered users. Each of these client computers monitors all changes made to cookies at that machine and sends any cookie change data to the Cookie Synchronization Server including the identity of the account to which the client computer belongs. In a preferred embodiment, the computer simply sends the entire cookie and an account ID with the request.

The Cookie Synchronization Server stores the information and subsequently sends it out to each other client machine of that account. The server may, for instance, send out the information to the registered client machines responsive to requests for the information from the other client machines. The client machines update their copies of the cookies accordingly and send an acknowledgement receipt to the Cookie Synchronization Server.

**The Howard Reference**

Howard pertains to a technique for simplifying for a computer user the accessing of web sites that require the user to register to access the website (e.g., requires a user ID and/or password). Specifically, a plurality of web sites register to be part of this service. The servers operated by those web sites are termed "affiliate servers". In addition, an "authentication server" is maintained.

When a user accesses an affiliate web server that requires authentication of the user (i.e., entry of a user ID and/or password), the affiliate server passes the client request to the authentication server instead of servicing it itself. (Col. 6, lines 55-57). The authentication server sends the client machine a sign-in page. (Col. 6, lines 59-62). When the user enters the proper password and ID, the authentication server copies certain cookies to the client machine and redirects the user's browser back to the affiliate server. (Col. 7, lines 16-20). These cookies comprise a first cookie that indicates the time that the user was authenticated and a second cookie containing the user's profile. (Col. 7, lines 23-27). These two cookies cannot be read by any affiliate

servers. (Col. 7, lines 40-41). The authentication server also generates an authentication ticket and transmits it to the affiliate server informing the affiliate server that the user has been properly authenticated. (Col. 7, lines 47-52). The authentication server also communicates the user profile information to the affiliate server through the client machines. (Col. 7, lines 58-67).

The authentication server also creates and maintains a cookie that contains a list of the affiliate servers visited by the used during a network session and then, when the used logs off the network, sends to each affiliate server on that list, a request for the affiliate server to delete any cookies it placed on the client computer system. (Col. 7, lines 27-38).

Howard has nothing to do with maintaining consistent copies of the same cookie across a plurality of different client machines. Howard merely achieves the goal of permitting a single user to be authenticated on multiple web sites with the same user name and password and by entering that user name and password only once per internet session.

**The Sears Reference**

In Sears, a number of web sites register with a server (called a "cookie server") and provide the server with information regarding what data fields are expected in cookies submitted from users/clients to that web site. These registered sites are listed in a cookie list stored at the cookie server and are provided to the client when the client logs into the cookie server. When a client subsequently initiates a connection to a web site within the cookie list, in addition to checking for any locally stored cookies, the client

will indicate to the cookie server that it is connecting to that web site. The cookie server then uses the cookie requirement information that it obtained from the web site, as well as user specific information, to generate one or more appropriate cookies that the web site may use and transmits the generated cookie(s) to the client. The client then uses those cookies to the web site in the normal course of navigating the web site so that the web site may provide a customized web page to the client.

Thus, instead of the user having to manually enter information so that the web site may generate cookie(s), the cookie(s) are automatically generated by the cookie server. This allows a user to type in information only once and then access different Web sites without having to re-enter the same information.

The client machine need not store the cookie. Instead, the automatically generated cookie for a particular web site may be stored at the cookie server for transmission to the client only when the client subsequently navigates to the web site.

For example, a user may have hundreds of cookies that are used when navigating to hundreds of different Web sites. If the user changes his home address, conventionally, the user would have to change his address information hundreds of times (i.e., once for each Web site). Sears solves this problem. Specifically, the user enters his new address once and transmits it to the cookie server. The cookie server would then update all of the cookies that the server is storing.

Sears does not mention anything about maintaining copies of the same cookie across a plurality of different client machines. Rather, Sears deals with maintaining consistency of information in different cookies in the same machine. Sears never

mentions a second user client machine. Sears' specification discusses a single user client machine.

**The JSOnline Reference**

JSOnline is an article discussing Web bugs, which some Web sites use to surreptitiously gather information about visitors, which is not particularly relevant to the present invention. However, the Office has focused on a single paragraph in the article that states "Using a web bug process called 'cookie sync,' two companies can exchange data in the background about Web site visitors. The information can be broad demographic data or personally identifiable elements and it's often for online profiling purposes, the Privacy Foundation says."

The Office asserts that this teaches the feature of synchronizing a cookie across said plurality of client computing devices containing a different copy of said cookie.

Nothing could be further from the truth. First, as in the other two cited references, there is only one client machine being discussed in the JSOnline article. There is absolutely no mention of multiple client machines. The quoted paragraph concerns sharing of information between two servers. Furthermore, although the technology discussed in this paragraph is called "cookie sync", it does not even involve synchronizing cookies (i.e., causing two copies of a cookie to be identical). It only involves sharing information about a user that might be found in a user cookie.

An exemplary use of "cookie sync", is as follows. Suppose a user surfs to abc123.com, which contains a Web bug. The Web bug looks like an image embedded

in the abc123.com Web page. In response to loading the abc123.com Web page, the user's computer automatically tries to fetch the embedded image. But the image is actually located at Bug.com. The user's client machine contacts Bug.com. Bug.com delivers an invisible image, so that the user is not even aware of it. Effectively, the abc123.com Web site has tricked the user's client machine into transmitting information from abc123.com and/or the user's client machine to Bug.com without the user's knowledge. Thus, Bug.com can potentially determine (1) the type of browser that fetched the Web bug image; (2) the time the Web bug was fetched; (3) the IP address of the computer that fetched the Web bug; (4) the URL of the main Web site; (5) the URL of the Web bug image located on the hidden Web site; and (6) a previously-set cookie value.

Thus, for example, if the user created an account at abc123.com in which the user provided her e-mail address. The Web bug could transmit that address to Bug.com without the user's knowledge.

Of course, the abc123.com Web site could have sent that information directly to Bug.com without involving the user's client machine. However, if abc123.com sent the information to Bug.com without going through the user's client machine, then Bug.com would know only that *someone* created a login at abc123.com with that e-mail address, but would have no way to associate the e-mail address with any particular client machine. However, by using a Web bug to transmit the information through the user's client machine to Bug.com, then Bug.com could identify the user's client machine.

As can be seen from the discussion above, despite the name, no actual synchronizing of cookies occurs at all, let alone across a plurality of client machines.

**The Cited References Teach None of the Claim Elements**

**The Elements of Exemplary Claim 1**

**Preamble**

The preamble recites "A method of synchronizing different copies of a cookie across a plurality of client computing devices that access a network".

The Office asserts that Howard pertains to this topic, but does not provide a specific citation within the reference. Howard does not pertain to synchronizing different copies of the same cookie across a plurality of different client machines. Rather, Howard pertains to a technique by which the user can enter necessary authentication information (e.g., user name and password) only once during a network session and be permitted access to multiple web sites requiring such authentication. Thus, in one sense, Howard pertains to a technique for permitting a single user at a single client machine to be authenticated to a plurality of different websites without having to enter a plurality of user IDs and passwords. This is simply a different topic. Howard contains no discussion of more than one client machine in an account

**Element (1)**

The first paragraph of claim 1 recites "(1) registering a plurality of client computing devices as members of an account, wherein at least one cookie is to be

synchronized across said plurality of client computing devices that are members of said account, each of said plurality of client computing devices containing a different copy of said at least one cookie".

The Office asserts that the first portion of this claim element, i.e., registering a plurality of client computing devices as members of an account, is taught in Howard at column 2, lines and 15-42 and column 5, lines 42-67 wherein the user of the client machine registers by providing necessary information to the authentication server.

These portions of Howard described that "the user of client computer system 100 and the operator of affiliate server 104 'register' with the authentication server 110". (Col. 5, lines 48-51)

Quite clearly, this is a discussion of a single client computer, not a plurality of client computers that are members of the "account". Accordingly, this portion of Howard does not disclose "registering a plurality of client computing devices as members of an account".

With respect to the portion of the claim that recites "wherein at least one cookie is to be synchronized across said plurality of client computing devices that are members of said account, each of said plurality of client computing devices containing a different copy of said at least one cookie", the Office asserts that this is found in JSOnline where it discloses using a web bug process called "cookie sync" so that two companies can exchange data in the background about website visitors. However, as described above in the discussion of the JSOnline reference, "cookie sync" merely concerns two servers sharing information that can be obtained from a cookie on the client machine. It has

nothing to do with sharing cookies per se, let alone sharing them amongst client machines.

**Element (2)**

The second paragraph of claim 1 recites “maintaining information identifying the members of said account at a server on said network”.

The Office asserts that this is found in column 10, line 55 through column 11, line 15 of Howard (which is claim 1 of Howard) and comprises the information received in the completed web page authentication information maintained by the authentication server.

This is not an accurate description of claim 1 of Howard. Claim 1 of Howard very nicely summarizes the technology disclosed in Howard. Obviously, the Office is referring to the fifth and sixth paragraphs of claim 1 which recite "communicating a web [page] from the authentication server to an Internet browser operated by the user, wherein the web page requests login information to be returned to the authentication server from the user" and "receiving the completed web page at the authentication server from the user".

First, the authentication (or login information) (i.e., a user name and password) in the returned, completed web page does not disclose the identity of the client machines that are members of the account. Secondly, there is only one member of the account. Howard never mentions multiple client machines forming a single account. Accordingly, this portion of Howard does not disclose this claim element.

In reviewing claim 1, Applicant noted that the claim previously recited maintaining information "as to the members of said account". Applicant has amended this language to now recite maintaining information "identifying the members of said account". Particularly, perhaps the Examiner was reading the terminology "maintaining information as to the members of said account" as reading on any information concerning a member of the account, such as a user name or password, whereas what Applicant actually was attempting to claim information that actually identifies all of the client machines in the account.

The new language should prevent any such unintended interpretation of the claim.

**Element (3)**

The third paragraph of claim 1 recites "responsive to a change in a copy of said at least one cookie stored at a first one of said client computing devices that is a member of said account, said first member client computing device sending a message to said server on said network, said message containing sufficient data from which said changes to said copy of said at least one cookie can be determined and the account to which said first member client computing device corresponds".

The Office asserts that this is found in Sears at column 3, lines 32-48 and column 10, line 51 to column 11 line 6. The cited portion of column 3 discloses the invention of Sears essentially as described above. Particularly, it recites in relevant part:

> The present invention allows the user to change the user information in each of these cookies by simply changing the user information stored at the cookie

> server. The cookie server would then update all of the corresponding cookies that the server is storing. Alternatively, the user would wait until a subsequent navigation to the website when the cookie server would generate a cookie with the new user information without requiring further user input. Thus, when a user's information changes, the user need not engaged in further data entry as the user navigates to each of the web sites. Instead, the user only changes the user information at the cookie server.

The cited portion in columns 10 and 11 discloses:

> In addition, the present invention allows the cookie to be more dynamic. For example, a user may have hundreds of associated cookies that are used when navigating to hundreds of different Web sites. If the user moves, for example, the user's address and zip code will change. Conventionally, in order to optimize service at these hundreds of web sites, the user would have to navigate through these web sites, and change the address for each web site, thereby allowing the web site to change the corresponding cookie at the client. The present invention allows the user to change the user information in each of these cookies by simply changing the user information stored at the cookie server. The cookie server would then update all of the corresponding cookies that the server is storing. Alternatively, the update would be postponed until a subsequent navigation to the web site when the cookie server would generate a cookie with the new user information without requiring further user input. Thus, when the user's information changes, the user need not engage in further data entry as the user navigates to each of the web sites. Instead, the user only changes the user information at the cookie server. This enhances the unified experience of the user since a change to user information for one Web site will be made for all affiliated Web sites.

These portion of Sears describe the user changing the information in the appropriate data field stored at the cookie server and then the cookie server placing that data into one or more different cookies at the cookie server. These portions of Sears differ from the claim element in a very significant way. Specifically, there is no cookie at the client machine that is being changed and to which the further processing is responsive, as claimed. In fact, in Sears, there are no cookies at the client machine at all. The whole point of Sears is that the client machine does not store cookies. Rather, the cookie server stores the data that is needed to generate data fields within cookies

and, when the client machine navigates to a web site listed in the cookie server's cookie list, only then does the cookie server build the appropriate cookie(s) for that particular web site and send it to the client machine.

Thus, these portions of Sears do not disclose any cookie being changed at a client machine that results in a message being sent to the cookie server, as claimed.

**Element (4)**

The fourth paragraph of claim 1 recites "storing said data at said server". The "data" referred to here is the data referenced in element (3) of claim 1, namely, the changes to the cookie at the client machine and the account ID.

The Office asserts that this is found in column 3, line 59 through column 4, line 2 of Howard in which the authentication server provides user profile information to the affiliate server.

This portion of Howard discloses:

> As part of the user authentication process, the authentication server 110 may provide certain user profile information to the affiliate server, such as the user's email address, user preferences, and the type of Internet browser installed on client computer 100. This user profile information is associated with the user's login ID so that each time the user logs into an affiliate server, the associated user profile information is available to provide to the affiliate server. This user profile allows the user to enter the information once and use that information during subsequent logins to new affiliate servers. (Col. 3, line 59 - col. 4, line 2).

Thus, this portion of Howard describes one server, namely, the authentication server, sending user information to another server, namely, the affiliate server and appears to be irrelevant to this claim element. On the other hand, Applicant does not dispute that, the authentication server as well as the affiliate server in Howard may store

user profile information. However, it is not cookie change data as defined in step (3) of the claim.

**Element (5)**

The fifth paragraph of claim 1 recites "said server sending said data to other client computing devices that are members of said account".

The Office asserts that this is found in Howard at column 7, lines 34-35, which describe the authentication server sending a message to each Web server on the list of sites visited. This portion of Howard states "For example, when the user logs out, the authentication server sends a message to each Web server on the list of sites visited. Each message is a request for the Web server to delete any cookies it placed on the client computer system (e.g., through a browser running on the client computer system)".

This feature of Howard of sending a message from one server, i.e., the authentication server, to another server, i.e., the affiliate server, asking the affiliate server to delete cookies when the user logs off does not have any relevance to the claimed feature of a server sending to the other client machines corresponding to a given account the changed cookie information. Specifically, (1) a request to delete cookies is not changed cookie information and (2) an affiliate server is not a client machine.

**Element (6)**

The sixth paragraph of claim 1 recites "each of said other client computing devices that is a member of said account updating its copy of said at least one cookie in accordance with said data".

The Office asserts that this is found in Howard at the same column 7, lines 25-39 referenced above in connection with element (5) and particularly in the disclosure that the authentication server also updates the cookie that contains a list of all sites visited by the user.

This portion of Howard discloses in relevant part:

> The authentication server also updates (or creates) a cookie that contains a list of all sites (or web servers) visited by the user since the last logout from the authentication server. The cookie is updated by adding the current affiliate server to the list of sites visited. This list of sites visited is used to remove cookies from the client computer system when the user logs out of the authentication server. For example, when the user logs out, the authentication server sends a message to each web server on the list of sites visited. Each message is a request for the web server to delete any cookies it placed on the client computer system (e.g., through a browser running on the client computer system).

This portion of Howard discusses two features, namely, (1) the authentication server maintaining a cookie listing the sites visited by the user and (2) the authentication server sending a message to affiliate servers asking them to delete cookies they placed on the client machine. Neither of these features has anything to do with client machines updating their cookie data. In this section of Howard, the cookie listing the visited sites is not sent to any other computer. This would not make any sense. This cookie only contains the list of affiliate servers to which the request to delete cookies must be sent. Furthermore, nothing at all is sent to any client machines in this portion of the Howard.

Even further, it is not seen how one could possibly interpret deleting a cookie as comprising updating a copy of a cookie in accordance with changed cookie data as that term is used in the claims of the present application.

**Summary With Respect to Claim 1**

Thus, it can be seen from the above discussion that the prior art of record actually does not disclose any element of claim 1, let alone all of them. This result is surprising since none of the three cited references has anything to do with sharing cookies across multiple client machines.

**The Remaining Claims**

Claims 2-15 depend from claim 1 and, therefore, distinguish over the prior art of record for at least all of the same reasons as claim 1.

The other independent claims, claims 16 and 24, contain similar recitations to claim 1.

Specifically, the preamble and steps (1) through (5) of independent claim 16 are almost identical in scope and language to the preamble and steps (1) through (5), respectively, of claim 1. The primary difference between claim 1 and claim 16 is that claim 16 is written specifically from the perspective of the server and, therefore, recites acts performed by the server, whereas claim 1 is a system claim reciting acts performed at both the client machine and the cookie

synchronization server. Claim 16 therefore distinguishes over the prior art of record for at least all of the same reasons as claim 1.

Claims 17 through 23 depend from claim 16 and, therefore, distinguish over the prior art of record for at least all of the reasons discussed above in connection with claim 16.

Independent claim 24 is written from the perspective of one of the client machines that is a member of the account. The preamble and steps (1), (2), and (4) of claim 24 very closely parallel the preamble and steps (1), (3), and (6), respectively, of claim 1. Therefore, claim 24 distinguishes over the prior art of record for at least all of the same reasons discussed above in connection with steps (1), (3), and (6) of claim 1.

Claims 25-32 depend from claim 24 and therefore distinguish over the prior art for at least all of the reasons set forth in connection with claim 24.

**Further Discussion**

Even further, regardless of all of the above, the proposed combinations are not suggested nor would any combination of the three references result in a system that synchronizes the same cookie at different client machines. None of the three applied prior art references even discusses more than one client machine, let alone has anything to do with synchronizing different copies of the same cookie across a plurality of client machines, which is the subject matter of the present claims. Therefore, no combination could possibly teach as much.

## Conclusion

In view of the foregoing remarks, this application is now in condition for allowance. Applicant respectfully requests the Office to issue a Notice of Allowance at the earliest possible date. The Examiner is invited to contact Applicant's undersigned counsel by telephone call in order to further the prosecution of this case in any way.

Respectfully submitted,

Dated: February 22, 2007

/Theodore Naccarella/

Theodore Naccarella
Reg. No. 33,023
Synnestvedt & Lechner LLP
2600 Aramark Tower
1101 Market Street
Philadelphia, PA 19107-2950
Telephone: (215) 923-4466
Facsimile: (215) 923-2189
Attorneys for Applicant

TXN:pmf

S:\I\IBM\IBM Raleigh RSW\Patents\P25025 USA\PTO\Response_to_OX_of_12.12.06(2) .doc